

AD-A110 121

GEORGIA INST OF TECH ATLANTA
COMPUTER AIDED SPECIFICATION TESTING SYSTEM (CASTS). VOLUME II. ETC(U)
MAR 81 D E HUMPHREY, D P MILLARD

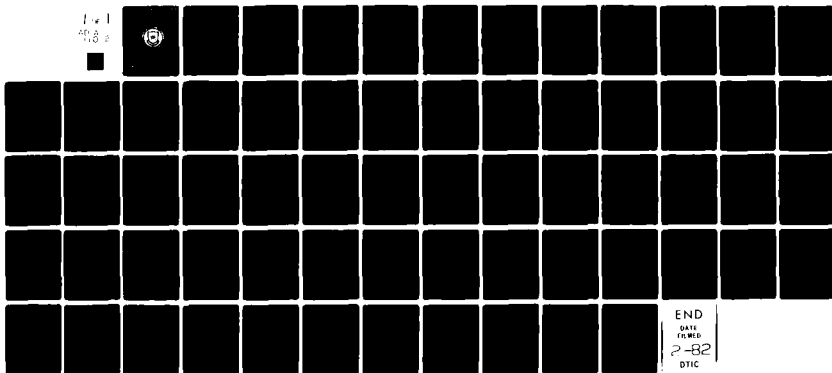
F/G 5/5

DAAK70-79-D-0087

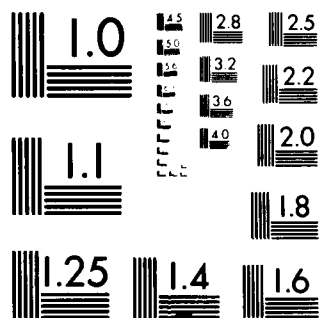
UNCLASSIFIED

ML

Fig 1
REV 8
11/8/8



END
DATE
FILMED
2-82
DTIC



MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

AIRMICS

III 20

AD A110121

US ARMY INSTITUTE FOR RESEARCH
IN MANAGEMENT INFORMATION AND COMPUTER SCIENCE



REC
JAN 27 1982

DTIC FILE COPY

CASTS
COMPUTER AIDED SPECIFICATION
TESTING SYSTEM
VOLUME II - USER'S MANUAL
MARCH 1981

Approved
for
distribution

01 26 82 039

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. 7D-414	3. RECIPIENT'S CATALOG NUMBER 0424
4. TITLE (and Subtitle) CASTS (USER'S MANUAL)		5. TYPE OF REPORT & PERIOD COVERED R&D Technical Report Final
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) D. E. Humphrey, D. P. Millard EES, Georgia Institute of Technology		8. CONTRACT OR GRANT NUMBER(s) DAAK70-79-D-0087 Task Order Number 0007
9. PERFORMING ORGANIZATION NAME AND ADDRESS Georgia Institute of Technology Atlanta, Georgia 30332		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS US Army Institute for Research in Management Information and Computer Science 115 O'Keefe Bldg., GIT Atlanta, Georgia 30332		12. REPORT DATE 15 March 1981
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		13. NUMBER OF PAGES 64
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE None
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release, distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) Same		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Computer-Aided Design, Man-Machine Interface, Human Factors, Specification Testing, Interactive, Data Entry.		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The Computer Aided Specifications Testing System (CASTS) is a tool to aid in the design of interactive systems. It is concerned with human factors and the man- machine interface during the initial capture of data and the editing of that data. CASTS is divided into two processes. The first process is that used by the designer to construct simulations of interactive video display terminal dialogue; the second process gathers the performance data of an actual user who runs the simulation designed in the first process.		

COMPUTER AIDED SPECIFICATION TESTING SYSTEM (CASTS)

USER'S MANUAL

EES/GIT Project A-2560

Prepared by

Computer Science and Technology Laboratory
Engineering Experiment Station
Georgia Institute of Technology
Atlanta, Georgia 30332

D.E. Humphrey
D.F. Millard

for

AIRMICS
Georgia Institute of Technology
Room 325, Hirman Research Building
Atlanta, Georgia 30332

under

DAAK70-79-D-0087
Task Order # 0007

Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A	

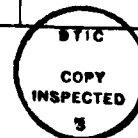


TABLE OF CONTENTS

	<u>Page</u>
1.0 INTRODUCTION	1
2.0 GENERAL SYSTEM USE	3
3.0 GETTING STARTED	8
3.1 Before Running CASTS	9
3.2 Login	10
3.3 Process I	10
3.4 Process II	10
4.0 PROCESS I SPECIFICATION	12
4.1 Preparation of Specifications	13
4.1.1 Transactions	13
4.1.2 Elements	14
4.1.3 Field Formats and Valid Values	15
4.2 Running Process I	17
4.3 Specification Command Language	18
4.3.1 Command Format	18
4.3.2 Item Names	19
4.3.3 Commands	19
4.4 Element Edit Criteria	23
4.5 Building a Sample Transaction	25
4.6 Specifying Screen Layouts	32
4.6.1 Screen Coordinates	32
4.6.2 Screen Areas	32
4.6.3 Screen Definition	34
4.7 Process I Error Messages	34
5.0 PROCESS I REPORTS	37
5.1 Data Element Dictionary	38
5.2 Transaction Dictionary	39
5.3 Transaction Screen Layout	40
6.0 PROCESS II SIMULATION	44
6.1 Running Process II	45
6.1.1 Run Sequence	45
6.1.2 Commands	50
6.1.3 Files Produced	50
6.1.4 Data Base Errors	50
6.2 Simulation Interaction Log	51
6.3 Process II Error Messages	51

TABLE OF CONTENTS

	<u>Page</u>
7.0 PROCESS II REPORTS	53
7.1 Terminal Session Error Report	54
7.2 Interaction Log Report	55
7.3 Application Raw Data Report	56

1.0 INTRODUCTION

1.0 INTRODUCTION

CASTS is a software system for use as a tool in the development of interactive data entry systems. It is composed of two major processes: (1) System Specification (Process I); and (2) System Simulation (Process II). The significant advantage of using CASTS is the capability for the system designer to return to the model construction phase of design to change specifications as often as necessary without major re-entry of the same information. Iterative refinements allow testing and modification of specifications without investing considerable coding effort in the initial system design process. In addition, the system designer utilizes an interactive tool which can provide valuable experience in designing other interactive systems.

CASTS was developed by the Computer Science and Technology Laboratory of the Engineering Experiment Station, Georgia Institute of Technology, for the U.S. Army Institute for Research and Management Information and Computer Sciences (AIRMICS). The concepts embodied in CASTS are an outgrowth of the research conducted by AIRMICS in the areas of human factors and the man-machine interface.

2.0 GENERAL SYSTEM USE

2.0 GENERAL SYSTEM USE

CASTS is a design tool for developing interactive data entry systems. It is divided into two processes. The first process is used by the system designer to construct simulations of interactive video display terminal dialogue, and the second process gathers the performance data of an actual user who runs the simulation designed in Process I of the system.

The Process I Specification is used interactively to describe or model the system to be developed. The specifications for the system may be sketchy and incomplete, or they may be very detailed. The Process II Simulation can then be run immediately to test the specifications under consideration. No coding by the system designer is required. Process I may be entered repeatedly to add detail or adjust the specifications depending on problems encountered in the simulation process. This cycle of refinement and test can continue until the system designer is satisfied with the system specifications (see Figure 1).

The basic element of the interactive system to be modeled is the transaction. A transaction may represent an employee record in a payroll application or a part record in an inventory application. A transaction consists of elements. Elements contain the actual data, such as employee identification number, employee name, or part number.

The elements are related to the transactions in a network fashion. Each transaction is composed of several elements. However, several transactions may share the same elements. These relationships are maintained within a data base management system.

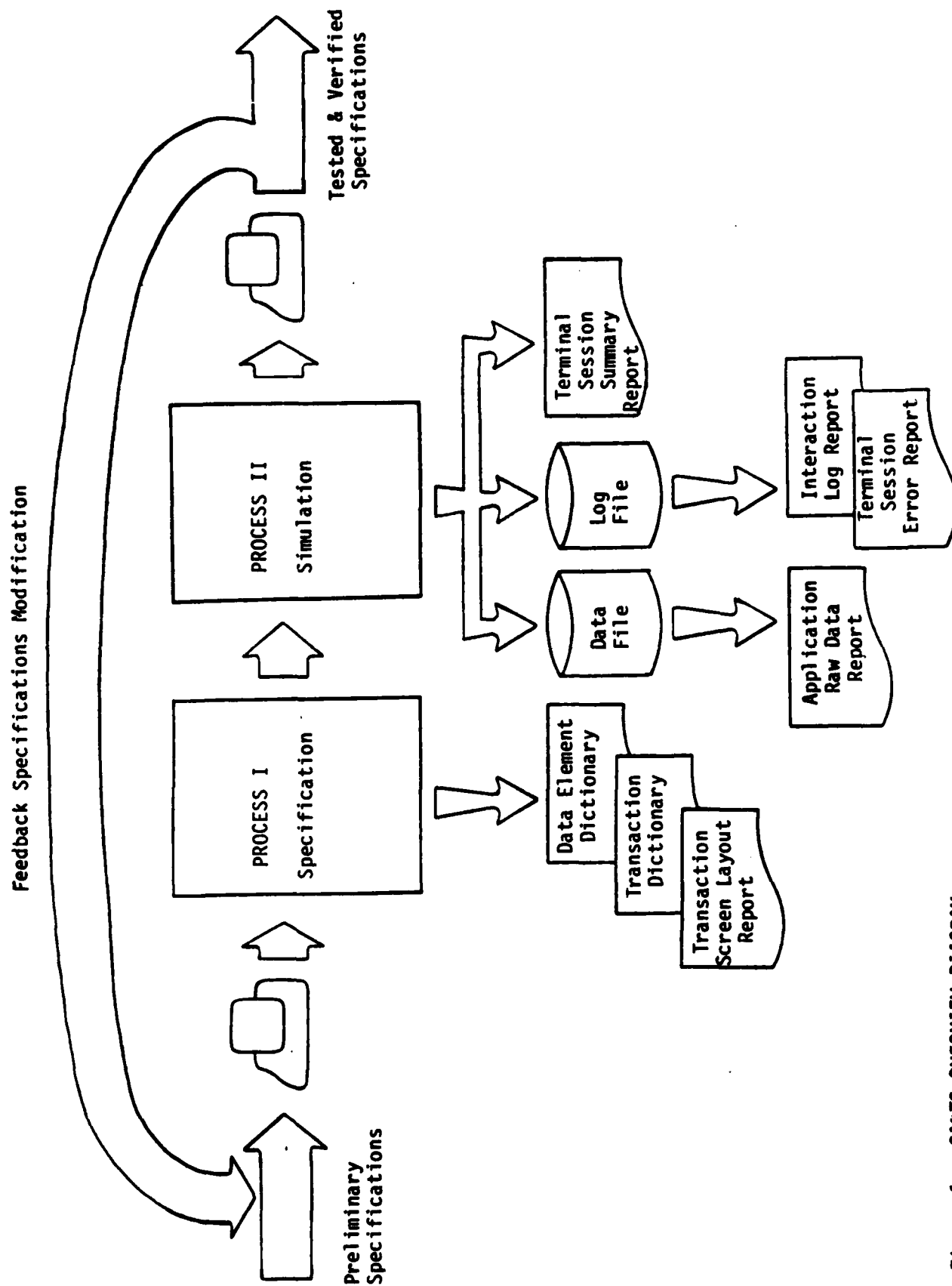
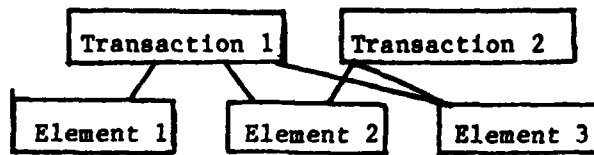


Figure 1. CASTS OVERVIEW DIAGRAM



During the Process I Specification the system designer interactively describes the application system transactions, element prompts, element edit criteria, and error messages. All or portions of this specification can be entered during the initial session. At any point in this dialogue, the system designer can terminate the session temporarily and receive reports detailing the current specifications or he may proceed to the Process II Simulation for system testing.

In the simulation process the specifications can be tested by entering sample transaction data in response to the element prompts. The transaction data entered will be checked in accordance with the edit criteria previously described. If no criteria has been specified for a particular element the simulation will accept any input, so as to allow incomplete specifications to be partially tested. The user of Process II may select one of several input modes for entering data, such as element by element prompting or a complete transaction at a time. During the simulation, a log of all user interaction with the system may be accumulated, time stamping all prompts, inputs and errors encountered. This log can be used to provide useful feedback information to the development process for the system under consideration.

A significant concept in CASTS is the capability for the system designer to return to the model specification phase of design to change specifications as often as necessary without major re-entry of the same

information. Iterative refinements allow testing and modification of specifications without investing considerable coding effort in the initial system design process. In addition, the system designer utilizes an interactive tool which can provide valuable experience in designing other interactive systems.

3.0 GETTING STARTED

3.0 GETTING STARTED

CASTS was designed and implemented on a DEC PDP-11/70 running under the IAS operating system. It utilizes a data base, DBMS-11, for the storage of the system specification information. The programs within CASTS are written in ANSI 1974 COBOL.

Throughout the user manual several notational conventions are followed. All prompts and messages from the computer system to the user are indicated by underlining. Expressions enclosed in angle brackets, < >, are to be replaced by user-supplied expressions. All others must be typed as shown in this manual.

3.1 BEFORE RUNNING CASTS

CASTS is designed to be a tool for designing interactive systems based on data entry into transactions. However, it is recommended that some preparation design work be done in advance of sitting down at a computer terminal.

The first step is to determine the kind of transaction to be built and what the constituent elements will be. As a minimum, a system designer must know the transaction name and have a list of the elements to be included. For each element, the system designer must know the name, the length of the element and whether or not it will be a constant field. Most elements will not be constant.

Each transaction has a transaction header, which contains information that applies to the transaction as a whole. One or more elements may be associated with a transaction. Each element has an element header, which contains information pertaining to the element. An element also has a prompt, edit criteria, error message, and help message. When the Process

II Simulation is run it will simulate an interactive system that accepts data input for a transaction. Each data field is an element and data is prompted for with the prompt specified in Process I. Data is checked according to the given edit criteria. Error messages and help messages can be displayed if specified in Process I.

3.2 LOGIN

The next step in using CASTS is to gain access to the computer system by getting an account from the system manager. To "login to" or "get on" the system, make sure that the terminal is on and, in response to the system prompt, type the following:

```
PDS> LOGIN
User name? <account name>
Password? <password>
```

NOTE: If the system prompt is not shown, type control C, written in this manual as CTRL C.

3.3 PROCESS I

To begin specifying elements and transactions, run CASTS, Process I.

```
PDS> RUN CASTS1
```

You will be asked to enter a user identification. This will be used to identify elements that you create and transactions that you build. There is an extensive on-line help facility that may be used to find information on how to use CASTS and what to enter in response to specific prompts. (See the HELP command in section 4.3.3 of this manual.)

3.4 PROCESS II

You may run CASTS Process II after you have specified transactions and

elements.

PDS> RUN CASTS2

Again, the first thing to be entered is a user identification. The <user id> entered here is used as the file name identifier. Following the simulation run, reports can be obtained from these files. The second question is whether to keep a time-stamped log of the user interaction with the simulation. An I/O mode must be chosen. New users will probably want prompt mode. At this point the name of a transaction can be entered and the simulation begins automatically.

4.0 PROCESS I SPECIFICATION

4.0 PROCESS I SPECIFICATIONS

4.1 PREPARATION OF SPECIFICATIONS

The minimum information that a system designer needs in order to use CASTS productively is what kind of transaction is to be built and what the constituent elements will be. Usually a system will consist of more than one transaction. Transactions would logically be grouped together into an application. For example, a personnel application might have transactions that represent new employees entering the system, pay rate changes for employees that get raises or change position, and assignment of employees to a position within a department.

In CASTS, transactions are stored as a transaction header, with information about the entire transaction, and the constituent elements that are associated with the transaction. All of the specifications for transactions and elements and the relationships among them are stored in a data base. However, the detailed workings of the data base are transparent to the user and need not concern him.

4.1.1 Transactions

The transaction header contains the name of the transaction, its length, an application name, a creation date, a creator id and information about screen boundaries for the screen mode of operation for Process II. The length and transaction creation information is handled automatically. The system designer should specify an application name, as some reports can be obtained which list transactions according to application. The screen boundary information does not need to be entered and discussion of the screen mode is deferred to a later section (see 4.6 Specifying Screen

Layouts).

In addition to transactions, the system designer initially needs to have some design for the elements involved. While building a transaction the information for the transaction header is requested, followed by prompts for the names of the elements that make up the transaction. These elements must either have already been specified or may be created during the building of the transaction.

4.1.2 Elements

Each element has an element header and is associated with a prompt message for that element, edit criteria, an error message, and a help message. The element header contains the name of the element, an element number or code, the element length, a description of the element, the proponent, a creation date, a creator id, and a constant field indicator. The element number is assigned by the system designer and can be a number or code that identifies the element. It may correspond to an element identifier in an already existing system or form. Although it is optional, it is recommended since some reports can be obtained based on the element number. The length of the element in character positions must be specified. A description of the element is optional and can be up to four lines long. The proponent is intended to specify the person or group responsible for keeping the element up-to-date. The creation date and creator id are handled automatically. The constant field indicator will usually be "N" for "NO". It exists so that an element can be specified to be a constant value, which means that it will not have data entered by the Process II user, but the default value will be entered automatically.

An element can have specified for it a prompt, edit criteria, error

message, and help message. The prompt is used by Process II to ask the Process II user to enter the input data for the element. If an element in an employee record is the employee's social security number, an appropriate prompt might be "ENTER SSN:".

The edit criteria for an element can be divided into two parts. The first is the format and default value. The second, available only if the format is numeric or alphanumeric, is range checks or value checks. For initial specifications, nothing is required to be entered for any of these.

The format for an element specifies the kind of input data that Process II will consider to be valid data. A format of all blanks will indicate that no data checking is to be performed. There are various format characters that indicate what each character position can be. For more information see the section 4.4 Element Edit Criteria. The default value will be what is entered for an element that the Process II user enters nothing for. Range checks and value checks are discussed in detail in section 4.4 Element Edit Criteria.

Error messages and help messages are optional. If no error message is specified for an element and the Process II user makes a mistake entering input data, only a system generated error message will appear. If no help message is specified and the Process II user requests help, he will be told that no help is available.

4.1.3 Field Formats and Valid Values

The following is a table that describes the valid input for the various specification fields.

Table 1. Field formats and Valid values

TRANS-REC	
TRANS-NAME	40 characters
APPL-NAME	x< 504, numeric
ENTRY-START-LINE	1< x < 20, numeric
ENTRY-END-LINE	1< x < 21, numeric, x >ENTRY-START- LINE
ERMSG-START-LINE	1< x < 22, numeric, x >ENTRY-END-LINE
ERMSG-END-LINE	1< x < 23, numeric, x > ERMSG-START-LINE
T-CREATE-DATE	6 digits
T-CREATOR-ID	9 characters, no special characters or all spaces.
TRANS-ELM-DEFN	
START-PSN-TRANS	X< 504, numeric
SCREEN-PROMPT-X	1< x < 72, numeric
SCREEN-PROMPT-Y	ENTRY-START-LINE < x < ENTRY-END-LINE, numeric
PROMPT-CONTENTS	72 characters
FIELD-X	1< x 72, numeric
FIELD-Y	ENTRY-START-LINE < x < ENTRY-END-LINE, numeric
ELEMENT-DEMO	
ELEMENT-NAME	30 characters
ELEMENT-DESC	288 characters
PROponent	40 characters
ELEMENT-NUMBER	6 characters
CONSTANT-FIELD	Y, N
ELEMENT-LENGTH	1< x < 72, numeric
E-CREATE-DATE	6 digits
E-CREATOR-ID	9 characters, no special characters or all spaces.
PROMPT-REC	
PROMPT-LINE	72 characters
EDIT-REC	
ELEMENT-FORMAT	Z,X,9,A,.,space, S
ELEMENT-DEFAULT	72 characters
ERROR-REC	
ERR-MSG	72 characters
HELP-MSG-REC	
ERR-MSG	72 characters
TRANS-LIST-REC	
TRANS-Name-L	40 characters
ELEM-LIST-REC	
ELEM-NAME-L	30 characters

4.2 RUNNING PROCESS I

CASTS Process I allows a user to specify elements and transactions to be used by Process II to simulate interactive data input into the transactions. Process I stores the information about the elements and transactions. Reports on the stored specifications can be obtained after running Process I by running the report programs.

To initiate Process I type in "RUN CASTS1" and hit RETURN. The program will display an introductory message. An important facility is the ability of the user to ask for help by typing "HELP" or "?". Beginning help is available under the topics of CASTS, HELP, START (how to get started) and COMMANDS.

The user will then be asked to enter his ID. The ID can be up to 9 characters long. It is used to identify transactions and elements that are created. The entry must be verified by typing a "Y", or "N" if it was mistyped.

The program next prompts for a command. Commands to specify elements and transactions include BUILD, CREATE, DELETE, DISPLAY, INSERT, MODIFY and REMOVE. The use and actions of these commands is discussed in Section 4.3 Specification Command Language. Process I performs each command, prompting for needed information. When a command is completed, the program displays "COMMAND>" and waits for the user to enter the next command.

Process I continues to respond to commands until the END command is entered after the "COMMAND>" prompt. At this point the entry of specifications is terminated. Process I reports can be run and/or Process II can be run.

4.3 SPECIFICATION COMMAND LANGUAGE

CASTS Process I provides a command language for the system designer to use in specifying transactions and elements. It allows for creating elements, building transactions, displaying stored specifications and changing existing specifications. The available commands are:

BUILD
CREATE
DELETE
DISPLAY
END
HELP
INSERT
MODIFY
REMOVE

4.3.1 Command Format

There are two command formats, one of which is a two-field format and the other is a three-field format. The positions or fields in the command string are delimited by slashes. If any position is null, i.e. there is nothing between the slashes, that position will assume the value from the previously entered command. If only the command is entered, the user will be prompted for the remaining information. The command formats are:

<command>/<item name>/<trans or elem name>
<command>/<trans or elem name or topic>

The <command> and <item name> are significant only to the first two letters and thus can be abbreviated.

The BUILD, CREATE, and HELP use the second or two-field command format. The others; DELETE, DISPLAY, MODIFY, REMOVE and INSERT use the three-field format. Several commands require more information and they will ask for it.

The fields of a command contain information that falls into three broad categories. The first is the command itself. The command name

always occupies the first field in the command string. The second category designates a transaction name, <trans name>; an element name, <elem name>; or a topic name, <topic name>. These occupy the second field of two-field commands and the third field of three-field commands. The third category is that of item name. An item name indicates a particular part of a transaction or element. Commands and item names are significant to and may be abbreviated to two letters.

4.3.2 Item Names

The item name occupies the second field of a three-field command. It allows the user to be more specific than just indicating an entire.

TRANS
SCREEN
ELEM
PROMPT
EDIT
ERROR
HELP
THDR
EHDR

The THDR and EHDR item names represent the transaction and element headers, respectively.

NOTE: Not all item names are valid for all commands.

4.3.3 Commands

BUILD/<trans name>

Use the BUILD command to name and build transactions. A transaction may be built from already existing elements or elements can be created during the building process.

CREATE/<elem name>

This command can be used to create individual data elements. It prompts the user for information associated with an element, such as a prompt, edit criteria, error message and help message.

DELETE/<item name>/<trans or elem name>

The DELETE command is used to delete an entire transaction, a whole element or an element prompt, edit criteria, error message or help message. To delete a transaction, TRANS is used as the item name. A transaction deletion removes all elements first and then deletes the transaction framework. ELEM is used as the item name when an element is to be deleted. Elements that belong to any transaction can not be deleted. To delete an element it must first be removed from all transactions.

If one line of a prompt, edit criteria, error message or help message is to be deleted, then the item name is followed by a line number in parentheses, called the item index. For example, to delete the second line of a prompt for an element type:

DELETE/PROMPT(2)/<elem name>

No item index is needed to delete the whole prompt, edit criteria, error message or help message.

Valid item names for the DELETE command are TRANS, ELEM, PROMPT, EDIT, ERROR and HELP.

Check for the "deletion complete" message. If it does not appear the deletion was not performed.

DISPLAY/<item name>/<elem or trans name>

This is a three-field command that allows the user to look at the specifications entered thus far. All of the item names except THDR and

EHDR are valid when matched with a corresponding element or transaction name. The display for a transaction includes the transaction header information and the elements in that transaction. To display the screen definition (screen position information) for an element, put the element name in the third command field. The transaction name will be prompted for.

END

The END command causes the user to exit the specification process (Process I). It may be issued only in response to the "COMMAND>" prompt.

HELP/<topic name>

The 'help' facility for CASTS consists of two methods of obtaining additional information. In response to the prompt "COMMAND>" you may enter the help command followed by the topic on which you are requesting information. The syntax is:

HELP/<topic name>

The other type of help is for use while entering specifications in Process I. In response to any prompt you may type a "?" for a explanation of the information to be entered, e.g.

ENTER ELEMENT NAME

?

To obtain help on other subjects before entering data:

?/<topic name>

At the end of each help message are suggestions for obtaining further related information.

INSERT/<item name>/<elem or trans name>

The INSERT command is used to insert an element into a transaction or to insert lines into a prompt, edit criteria, error message or help

message. To insert an element into a transaction, use the item name ELEM and the name of the transaction to insert the element into. You will then be prompted for the name of the element to be inserted and the name of the element to insert it after.

To insert a line, use the item name followed by the line number, called the item index, in parentheses, i.e. PROMPT(2). You may insert an element at the start of the list of elements in a transaction by entering only a carriage return in response to the question "after which element?". You may insert a line of a message at the beginning by giving an item index of zero. Also, no item index given is assumed to mean insert at the beginning, except for edit lines where the format and default line must be first.

The valid item names for this command are ELEM, PROMPT, EDIT, ERROR and HELP.

MODIFY/<item name>/<elem or trans name>

MODIFY is a three-field command that allows the user to change the specifications that have been entered. The item names TRANS, ELEM and LIST are invalid as they have no meaning in this command. However, THDR, SCREEN, EHDR, PROMPT, EDIT, ERROR and HELP are valid item names.

The prompt, edit criteria, error message and help message may be modified by putting the line number in parentheses after the item name as the item index. For example, to modify the third help line of a help message for an element:

MODIFY/HELP(3)/<elem name>

REMOVE/ELEM/<trans name>

This command is used to remove an element from a transaction. The item name in the second command field must be ELEM. The <trans name>

is the transaction from which the element is to be removed. The name of the element to be removed is asked for after the command is typed in.

An element must be removed from any transactions it is a part of before it can be deleted.

Table 2. Summary of Valid Item Names for Each Command

	TR	TH	SC	EL	EH	PR	ED	ER	HE	LI
BUILD	(X)									
CREATE				(X)						
DELETE	X			X		X	X	X	X	
DISPLAY	X		X	X		X	X	X	X	X
MODIFY		X	X		X	X	X	X	X	
REMOVE				X						
INSERT				X		X	X	X	X	

4.4 ELEMENT EDIT CRITERIA

The edit criteria for an element is a set of criteria that indicate to Process II how to perform data checking on data entered for the element. For each element there must be a format and a default value, even if it is spaces, meaning anything is acceptable. In addition, alphanumeric elements may have value checks specified i.e., a list of valid data values. Numeric elements may have specified either a list of valid values or an upper and lower bound for a range check.

The edit criteria are grouped so that the user may access them more easily. The grouping may be thought of as 'lines' of edit information. An example for an element with a list of valid values would be organized as:

1)<format> <default>

2)<edit type><value>

3)<edit type><value>

4)<edit type><value>

with as many 'lines' of edit type and value as the user wishes. The edit type indicates whether the <value> is a value in a list of valid values or an upper bound or a lower bound. The organization for a numeric element with a range of valid values would be:

1) <format> <default>

2) <edit type><lower bound>

3) <edit type><upper bound>

An actual example may be instructive. Assume that there is a numeric element whose value must be greater than or equal to one and less than or equal to 100, and it has a default value of 50. The edit lines would be:

1) 999 050

2) LOWER BOUND 001

3) UPPER BOUND 100

All elements must have a format and default specified and thus must have the first line of edit information. In the DELETE, MODIFY and INSERT commands the line in question is indicated by the item index.

The element format describes the allowable data input characters for each character position of an element. There are six format characters.

Each character position must have conforming to the format given:

- 9 Numeric (0-9)
- A Alphabetic (-Z)
- X Alphanumeric (0-9,A-Z)
- Z Zero suppression (0,space)
- . Decimal Point
- S Sign (+,-,space)

For example, an element format of 'ZZ9.99' means that the element is

numeric with two digits to the right of the decimal point and if less than 100.00 the leading zeroes may be replaced by spaces: ' 9.65' instead of '009.65'. Any character other than these is assumed to be a special character and the value in that position must be that special character. If no format is specified, then data entered for the element will not be checked.

The element default value is the value for an element to be entered automatically by Process II as input data if no input is entered by the user. It must satisfy the element format.

The edit type indicates the type of the value it is associated with. The edit type will be 'VALUE', 'LOWER BOUND' or 'UPPER BOUND'.

The edit value, associated with an edit type, will be interpreted as a valid value, lower bound or upper bound depending on the edit type. The type. The edit value must conform to the element format.

4.5 BUILDING A SAMPLE TRANSACTION

The following pages show the user interaction with Process I in building a transaction named TESTTRANS. The transaction has four elements. Three elements were created individually. Then the transaction was built. The third element could not be found, so it was created during the build process. Logically, the transaction looks like the following diagram. The format of each element is shown in the character positions.

TESTTRANS

ELEM1				ELEM2								ELEM3		ELEM4									
9	9	9	9	A	A	A	A	A	A	A	A	A	X	X	9	9	9	-	X	X	-	A	A
1				4	5								14	15	16	17							25

PGS. RUN PAGE 1
L: 08:06

***** COMPUTER AIDED TECHNOLOGICAL DESIGN SYSTEM *****
PROCESS I: GET HELP TOP
AIRMAILS GET

WELCOME TO CASTS!

IF YOU WOULD LIKE HELP, PLEASE TRY:

HELP/CASTS - FOR A DESCRIPTION OF THE CASTS SYSTEM
HELP/HELP - FOR INFORMATION ON THE 'HELP' FACILITY
HELP/COMMANDS - FOR A LIST OF AVAILABLE COMMANDS
HELP/START - HOW TO GET STARTED IN PROCESS I

FIRST, PLEASE ENTER YOUR ID (UP TO 9 CHARACTERS):

DH

IS THIS CORRECT: DH TYPE 'Y' OR 'N'

Y

COMMAND> HELP/START

TO GET STARTED USING THE CASTS PROCESS I YOU WILL WANT TO
CREATE ELEMENTS AND BUILD TRANSACTIONS. ELEMENTS SHOULD EXIST BEFORE
THEY ARE BE BUILT INTO TRANSACTIONS. USE THE 'CREATE' COMMAND TO
CREATE AN ELEMENT AND YOU WILL BE PROMPTED FOR INFORMATION ABOUT
THE ELEMENT, THE PROMPT YOU WANT FOR IT AND ANY EDIT CRITERIA,
ERROR MESSAGES OR HELP MESSAGES YOU MAY WISH TO SPECIFY. A BLANK
LINE TERMINATES THE INPUT.

USE THE 'BUILD' COMMAND TO BUILD TRANSACTIONS. YOU WILL
BE ASKED TO ENTER INFORMATION ABOUT THE TRANSACTIONS AND TO LIST
THE ELEMENTS TO BE INCLUDED. EXPERIENCED USERS MAY OPTIONALLY
SPECIFY INFORMATION RELATING TO A SCREEN MODE OF INPUT FOR USE
IN PROCESS II.

THE 'DISPLAY' COMMAND ALLOWS YOU TO CHECK THE SPECIFICATIONS
ENTERED AND THE 'MODIFY' COMMAND MAKES IT POSSIBLE TO CHANGE THEM.
USE THE 'HELP' COMMAND OR '?' ANYTIME YOU ARE UNCERTAIN AS TO WHAT
TO ENTER.

COMMAND> CREATE/ELEM1

PLEASE ENTER ELEMENT HEADER INFORMATION:

ENTER ELEMENT DESCRIPTION (4 LINES)

DESC OF ELEM1, LINE1

DESC OF ELEM1, LINE2

ENTER PROPONENT

FIRST PROPONENT

ENTER ELEMENT NUMBER

000001

ENTER CONSTANT FIELD INDICATOR

N

ENTER ELEMENT LENGTH

4

HEADER COMPLETE FOR ELEMENT: ELEM1

PLEASE ENTER PROMPT LINES:

ENTER PROMPT LINE

ENTER ELEM1 >

ENTER PROMPT LINE

PLEASE ENTER EDIT INFORMATION:

ENTER ELEMENT FORMAT

9999

ENTER ELEMENT DEFAULT VALUE

0001

NUMERIC ELEMENTS MAY HAVE RANGE OR VALUE CHECKS.

RANGE CHECKS, VALUE CHECKS OR NEITHER? (R/V/N)

N

PLEASE ENTER ANY ERROR MESSAGE LINES:

ENTER ERROR LINE

ERROR MSG FOR ELEM1, LINE1

ENTER ERROR LINE

ERROR MSG FOR ELEM1, LINE2

ENTER ERROR LINE

ERROR MSG FOR ELEM1, LINE3

ENTER ERROR LINE

PLEASE ENTER ANY HELP MESSAGE LINES:

ENTER HELP LINE

HELP MSG FOR ELEM1, LINE1

ENTER HELP LINE

ELEMENT CREATION COMPLETE: ELEM1

COMMAND> CREATE/ELEM2

PLEASE ENTER ELEMENT HEADER INFORMATION:

ENTER ELEMENT DESCRIPTION (4 LINES)

DESC OF ELEM2, LINE1

ENTER PROPONENT

FIRST PROPONENT

ENTER ELEMENT NUMBER

000002

ENTER CONSTANT FIELD INDICATOR

N

ENTER ELEMENT LENGTH

10

HEADER COMPLETE FOR ELEMENT: ELEM2

PLEASE ENTER PROMPT LINES:
ENTER PROMPT LINE
ENTER ELEM2 >
ENTER PROMPT LINE

PLEASE ENTER EDIT INFORMATION:
ENTER ELEMENT FORMAT
AAAAAAAAAAAA
ENTER ELEMENT DEFAULT VALUE

ALPHA/ALPHANUMERIC ELEMENTS MAY HAVE VALUE CHECKS.
ENTER EDIT VALUE

PLEASE ENTER ANY ERROR MESSAGE LINES:
ENTER ERROR LINE
ERROR MSG FOR ELEM2, LINE1
ENTER ERROR LINE

PLEASE ENTER ANY HELP MESSAGE LINES:
ENTER HELP LINE
HELP MSG FOR ELEM2, LINE1
ENTER HELP LINE
HELP MSG FOR ELEM2, LINE2
ENTER HELP LINE

ELEMENT CREATION COMPLETE: ELEM2

COMMAND> CREATE/ELEM4

PLEASE ENTER ELEMENT HEADER INFORMATION:
ENTER ELEMENT DESCRIPTION (4 LINES)
DESC OF ELEM4, LINE1
DESC OF ELEM4, LINE2
DESC OF ELEM4, LINE3
DESC OF ELEM4, LINE4
ENTER PROPONENT
SECOND PROPONENT
ENTER ELEMENT NUMBER
000004
ENTER CONSTANT FIELD INDICATOR
N
ENTER ELEMENT LENGTH
9
HEADER COMPLETE FOR ELEMENT: ELEM4

PLEASE ENTER PROMPT LINES:
ENTER PROMPT LINE
ENTER ELEM4 >
ENTER PROMPT LINE

PLEASE ENTER EDIT INFORMATION:

ENTER ELEMENT FORMAT

999-XX-AA

ENTER ELEMENT DEFAULT VALUE

000-B6-MM

PLEASE ENTER ANY ERROR MESSAGE LINES:

ENTER ERROR LINE

ERROR MSG FOR ELEM4, LINE1

ENTER ERROR LINE

PLEASE ENTER ANY HELP MESSAGE LINES:

ENTER HELP LINE

HELP MSG FOR ELEM4, LINE1

ENTER HELP LINE

ELEMENT CREATION COMPLETE: ELEM4

COMMAND> BUILD/TESTTRANS

PLEASE ENTER TRANSACTION HEADER INFORMATION:

ENTER APPLICATION NAME

TEST APPLICATION

DO YOU WISH TO HAND-TAILOR A SCREEN FORMAT?

NOTE: THIS REQUIRES SPECIAL KNOWLEDGE.

PLEASE SEE USER MANUAL, SECTION 4.6.

SCREEN FORMAT? (Y/N)

N

HEADER COMPLETE FOR TRANSACTION: TESTTRANS

PLEASE ENTER THE ELEMENTS FOR THIS TRANSACTION:

NOTE: AN EMPTY LINE INDICATES END OF INPUT

ENTER ELEMENT NAME

ELEM1

ENTER START POSITION IN TRANSACTION

1

ENTER ELEMENT NAME

ELEM2

ENTER START POSITION IN TRANSACTION

5

ENTER ELEMENT NAME

ELEM3

CA003: ELEMENT BY THIS NAME DOES NOT EXIST

IF YOU MISTYPED THE NAME, AND WISH TO

REENTER, TYPE 'R'. TO CREATE THE

ELEMENT NOW, TYPE 'C'.

REENTER OR CREATE? (R/C)

C

PLEASE ENTER ELEMENT HEADER INFORMATION:
ENTER ELEMENT DESCRIPTION (4 LINES)
DESC OF ELEM3, LINE1

ENTER PROPONENT
SECOND PROPONENT
ENTER ELEMENT NUMBER
000003
ENTER CONSTANT FIELD INDICATOR
N
ENTER ELEMENT LENGTH
2
HEADER COMPLETE FOR ELEMENT: ELEM3

PLEASE ENTER PROMPT LINES:
ENTER PROMPT LINE
ENTER ELEM3 >
ENTER PROMPT LINE

PLEASE ENTER EDIT INFORMATION:
ENTER ELEMENT FORMAT
XX
ENTER ELEMENT DEFAULT VALUE
RS
ALPHA/ALPHANUMERIC ELEMENTS MAY HAVE VALUE CHECKS.
ENTER EDIT VALUE

PLEASE ENTER ANY ERROR MESSAGE LINES:
ENTER ERROR LINE
ERROR MSG FOR ELEM3, LINE1
ENTER ERROR LINE

PLEASE ENTER ANY HELP MESSAGE LINES:
ENTER HELP LINE
HELP MSG FOR ELEM3, LINE1
ENTER HELP LINE

ELEMENT CREATION COMPLETE: ELEM3

ENTER START POSITION IN TRANSACTION
15

ENTER ELEMENT NAME
ELEM4

ENTER START POSITION IN TRANSACTION
17

ENTER ELEMENT NAME

TRANSACTION BUILD COMPLETE: TESTTRANS

COMMAND> DISPLAY/TRANS/TESTTRANS

CURRENT TRANSACTION LENGTH

0025

CURRENT APPLICATION NAME

TEST APPLICATION

CURRENT TRANSACTION CREATE DATE

810306

CURRENT TRANSACTION CREATOR ID

DH

ELEMENT(S) IN THIS TRANSACTION:

	START	LENGTH
ELEM1	1	4
ELEM2	5	10
ELEM3	15	2
ELEM4	17	9

COMMAND> END

13:00:15 SIZE: 29K CPU: 6.76 STATUS: SUCCESS

FDS>

4.6 SPECIFYING SCREEN LAYOUTS

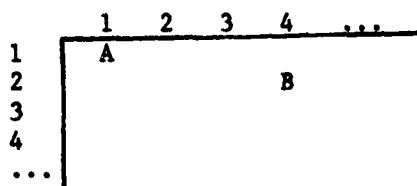
The ability to specify screen layouts for data input to transactions is an advanced feature of CASTS. It allows the system designer - Process I user - to describe screen formats for positioning elements of a transaction on the CRT screen. Hence, a data entry system can be designed to match source data forms for ease and efficiency in performing the data entry. Screen layout specifications, or screen definitions, are not necessary to be able to use CASTS. In building transactions, the user may choose not to specify a screen and this choice is recommended for novice users. However, laying out screens is a skill worth learning due to the benefits of using the screen mode of transaction data entry in Process II.

In the screen mode of data entry in Process II, the prompts for each of the elements are printed on the screen. The cursor then moves to the data entry field of the first element. After data is entered successfully, the cursor moves automatically to the next element. The screen is divided into two areas, although no division line or marker is displayed. In the upper user interaction area the element prompts are placed and data is entered in the data input fields. In the lower error message area messages are displayed when the user enters erroneous input or requests help.

4.6.1 Screen Coordinates

A CRT screen is usually 24 lines with 80 characters per line. For CASTS, 23 of the 24 lines may be used and 72 of the 80 characters. Positions on the screen are indicated with a coordinate pair (x,y). The x is the number of characters across a line (horizontal). Character positions start at 1 on the left side and increase toward the right. The y is the number of lines. Line numbers start at 1 on the top and increase

downward.



For example, in the above diagram, A is at position (1,1) and B is at position (4,2).

Screen coordinates are used to specify where on the screen to position the first character of the prompt and of the data field.

4.6.2 Screen Areas

The screen is divided into two areas, which are named:

1. User interaction area
2. Error message area

Data entry is accomplished in the upper user interaction area and the lower error message area is used for error and help messages. To specify the size of the two screen areas, a starting and ending line is required for each.

The screen area boundaries are specific to each transaction, and may be different for each transaction. Thus they are specified in the transaction header (THDR). If the user chooses to specify a screen, he will be prompted for the entry start line, entry end line, error message start line and error message end line. Table 1 in section 4.1.3 Field Formats and Valid Values shows the values that can be entered for these.

NOTE: Process I puts 99 as the entry start line when no screen areas and screen definition are specified.

4.6.3 Screen Definition

The screen definition (SCREEN) specifies the position of the element prompt on the screen, an element prompt and the position of the data field on the screen. The element prompt position is requested as the screen prompt position (x) and the screen prompt position (y). The screen prompt contents allows the system designer to specify a screen prompt different from the original element prompt. Whereas, the original prompt may be multi-line the screen prompt is restricted to one line and presumably will be short and concise. The element data will be entered at a position specified by the data field position (x) and data field position (y). Also included in the SCREEN item and not directly related to the screen definition is the start position of the element within the transaction.

4.7 PROCESS I ERROR MESSAGES

CA001: INVALID COMMAND

The command entered was invalid. The valid commands are BUILD, CREATE, DELETE, DISPLAY, MODIFY, REMOVE, INSERT, HELP, and END. Only the first two letters of each command are used, and the remaining letters are insignificant. For more information, see User's Manual Section 4.3.

CA002: TRANSACTION BY THIS NAME DOES NOT EXIST

There is no transaction specified with the name given. Check that the transaction name was typed correctly. Common mistakes that result in this message are typographical errors and giving an element name instead of a transaction name.

CA003: ELEMENT BY THIS NAME DOES NOT EXIST

There is no element specified with the name given. Check that the element name was typed correctly. Common mistakes that result in this message are typographical errors and giving a transaction name instead of an element name.

CA004: INVALID ITEM GIVEN

The item name entered in the second field of the command was invalid. The valid item names are TRANS, THDR, (transaction

header), ELEM, EHDR (element header), PROMPT, EDIT, ERROR, HELP, LIST, and SCREEN. Only the first two letters of each item name are used and the remaining letters are insignificant. For more information, see User's Manual Section 4.3.

CA005: UNKNOWN SYSTEM ERROR - SEE ANALYST

A return code internal to the program (RETURN-STATUS) has become an unexpected value (i.e., other than 000 to 006). Please write down what you were doing and see the CASTS System Administrator.

CA006: INVALID OPTION

The three letter option given was invalid. The option entered must be one of those displayed just prior to the "ENTER OPTION" prompt, and it must be typed exactly as shown. If both of these were done correctly and the message still appears, write down what you were doing and see the CASTS System Administrator.

CA007: TRANSACTION WITH THIS NAME ALREADY EXISTS

A transaction already exists with the transaction name given. There cannot be more than one transaction built with the same name. Display the transaction, and check if you have already built the transaction. If not, try a different transaction name.

CA008: ELEMENT WITH THIS NAME ALREADY EXISTS

An element already exists with the element name given. There cannot be more than one element created with the same name. Display the element and check if you have already created the element and/or wish to use it as is. If not, try a different element name.

CA009: NO HELP AVAILABLE

There is no help available for the topic requested.

CA010: ONLY THE CREATOR IS AUTHORIZED TO DELETE

The user id of the person currently running CASTS1 must match the creator id of an element or a transaction in order to be able to delete it.

CA011: ITEM NAME ILLEGAL FOR THIS COMMAND

The item name given may be valid; however, it cannot be used in conjunction with the command given. Use the HELP command with the name of the command in question to find out what item names are valid for that command, e.g., HELP/<command>.

CA012: ELEMENT NOT DELETED

The request to delete an element has not been completed. An element must be removed from all transactions before it can be successfully deleted.

CA013: INVALID VALUE FOR THIS FIELD

The value given does not satisfy the criteria for this data field. Use the HELP command (HELP or ?) to get information about what should be entered. For more information, see User's Manual Section 4.1.

CA014: ELEMENT NOT USED IN THIS TRANSACTION

The element name given is that of an element that does not belong to the transaction given.

CA015: DATA BASE ACCESS ERROR

An error has been made in accessing the stored specifications. Check to see how much of the operation that was in progress was completed. Try to recover by making modifications to complete the operation. If unrecoverable, see the CASTS System Administrator.

CA016: ITEM NOT FOUND

This message appears when an item requested cannot be found in the stored specifications. If the item requested was a prompt, edit range or value check, error message or help message, there may not be one for the element given. If a particular line of an item was requested, that item may not have a line with the given line number.

CA017: ITEM INDEX INVALID

The item index given was invalid. Either it contains illegal characters in an illegal format, or it refers to a line number of an item, such as a prompt, where that item does not have a line with the given line number.

CA018: INVALID CHARACTER DETECTED IN INPUT

Non-printable and certain special characters are not allowed as input. The following special characters are valid: ! " # \$ % & ' () - = / \ ? . , < > ; + : @ []

5.0 PROCESS I REPORTS

5.0 PROCESS I REPORTS

The Process I reports provide printed information on the current specifications as they are stored. It is recommended that current copies of these reports be kept on hand to provide a quick reference for the system designer.

5.1 DATA ELEMENT DICTIONARY

The Data Element Dictionary (ELERPT) details the data elements and their associated information. It allows the user to choose the category of elements to be reported and the order in which they will appear.

The Data Element Dictionary program is initiated by the Command:

RUN ELERPT

The user interaction sequence is as follows:

1. The user is presented with a menu of the categories of elements that can appear on the report and is requested to select one. The choices are:
 - A. All elements.
 - B. A range of element numbers.
 - C. Elements of a proponent.
 - D. Elements in an application.
2. The user is presented with a second menu of orders in which elements may appear. The user may select either:
 - A. Numeric by element number, or
 - B. Alphabetic by element name.
3. If for the first menu option two was chosen, the user is queried for a range of element numbers. If option three is chosen, the user is requested to enter a proponent. If option four is chosen, the user is requested to enter an

application name.

4. Processing commences and the elements are located, sorted, and formatted into report form.
5. The user may obtain the report by printing the file "ELERPT.TMP".

5.2 TRANSACTION DICTIONARY

The Transaction Dictionary (TRARPT) reports on the transactions and information related to them. It allows the user a choice as to the category of transactions to appear on the report. The transactions are reported in alphabetical order by transaction name.

The Transaction Dictionary program is initiated by the command:

RUN TRARPT

The user interaction sequence is as follows:

1. The user is presented with a menu of the categories of transactions that can appear on the report and is requested to select one. The choices are:

A. All transactions, or
B. Transactions in one application.

If the second option is chosen, the user is asked to enter the application name.

2. Processing commences and the transactions are located, sorted, and formatted into the report form.
3. The report may be obtained by printing the file "TRARPT.TMP".

5.3 TRANSACTION SCREEN LAYOUT

The Transaction Screen Layout (SCRRPT) is a report that will provide a printed version of the screen layout of individual transactions, i.e., the screen as it would appear during Process II in the screen mode of interaction. The screen layout specifications are reported, showing the transaction name, application, and the placement of element prompts and their associated data fields. Screen displays are reported in alphabetical order by transaction name. Some transactions may not have a specification for a screen layout.

The Transaction Screen Layout program is initiated by the command:

RUN SCRRPT

The user interaction sequence is as follows:

1. The user is presented with a menu of the categories of elements that can appear on the report and is requested to select one. The options are:
 - A. All transactions.
 - B. Transactions in an application.
 - C. One transaction.

If the second option is chosen, the user is asked to enter an application name. If the third option is chosen, the program requests a transaction name.

2. Processing commences and the transactions are located, sorted, and formatted into report form.
3. The report may be obtained by printing the file "SCRRPT.TMP".

COMPUTER AIDED SPECIFICATION TESTING SYSTEM

PAGE 1

DATE 11/03/06

AIRMC'S GIT

DATA ELEMENT DICTIONARY

ALL ELEMENTS IN DATA BASE

ELEMENT NAME-----NUMBER--LENGTH--CREATE DATE--CREATOR ID--CONSTANT--PROponent--
ACCOUNT NUMBER CS-003 0001 80/11/25 DAN NO COMPUTER SCIENCE DIVISION

FORMAT: 99999

DEFAULT:

DESCRIPTION: ACCOUNT NUMBER IS A NUMBER FROM 1 TO THE MAXIMUM NUMBER OF ACCOUNTS PERMITTED IN THE SYSTEM. IT IS ASSIGNED BY THE SYSTEM DURING THE PROCESSING OF A TRANSACTION TO ESTABLISH THE ACCOUNT. IT MUST BE USED IN ALL TRANSACTIONS INVOLVING CHANGES TO ACCT STATUS

TRANS X REF: NEW DEPOSIT
DEPOSIT
CHECK

ELEMENT NAME-----NUMBER--LENGTH--CREATE DATE--CREATOR ID--CONSTANT--PROponent--
BENT LEG SIT UP 40 0003 81/01/21 GLENN NO AIRMC'S

FORMAT: 999

DEFAULT: 000

DESCRIPTION: NUMBER OF BENT LEG SIT UPS COMPLETED

TRANS X REF: BPT

ELEMENT NAME-----NUMBER--LENGTH--CREATE DATE--CREATOR ID--CONSTANT--PROponent--
CADET NUMBER 20 0004 81/01/21 GLENN NO AIRMC'S

FORMAT: 9999

DEFAULT: 9999

DESCRIPTION: CADET PERSONNEL NUMBER

TRANS X REF: AGC
BPT

ELEMENT NAME-----NUMBER--LENGTH--CREATE DATE--CREATOR ID--CONSTANT--PROponent--
CHECK NUMBER CS-001 0004 80/11/25 DAN NO AIRMC'S

FORMAT: 9999

DEFAULT:

LOWER BOUND: 0

UPPER BOUND: 9999

DESCRIPTION: CHECK NUMBER IS THE IDENTIFYING ELEMENT FOR ALL CHECKS FOR A PARTICULAR ACCOUNT. IT MUST BE UNIQUE WITHIN A GIVEN MONTH AND IN THE RANGE FROM 0 TO 9999. ONLY NUMERIC CHARACTERS ARE ALLOWED.

TRANS X REF: NEW DEPOSIT
CHECK

ELEMENT NAME-----NUMBER--LENGTH--CREATE DATE--CREATOR ID--CONSTANT--PROponent--
OFFER 000124 0005 81/01/05 SAM NO CSTL

FORMAT: 99999

DEFAULT: 55555

VALID VALUE: 55555

VALID VALUE: 22222

VALID VALUE: 11111

COMPUTER AIDED SPECIFICATION TESTING SYSTEM

PAGE 1

DATE 01/01/00

AIMICS GLT

TRANSACTION DICTIONARY

ALL TRANSACTIONS IN DATA BASE

TRANSACTION NAME-----APPLICATION-----LENGTH-----CREATOR-----
AGI AGC 16 01/01/22 DAVID

-----ELEMENTS-----

UNIT ASSIGNED 10
PROMPT: PLEASE ENTER UNIT ASSIGNED TO
4-POSITION ALPHA-NUMERIC
ERROR MESSAGE: SYNTAX ERROR
THIS FIELD IS 4 CHARACTERS A/N

CADET NUMBER 20
PROMPT: ENTER CADET NUMBER(9999)
ERROR MESSAGE: THIS FIELD IS A 4POSITION NUMERIC

STATUS 222
PROMPT: ENTER STATUS
ERROR MESSAGE: ERROR STATUS
HELP MESSAGE: STATUS FORMAT "X"

SOURCE CODE 5
PROMPT: ENTER SOURCE CODE
ERROR MESSAGE: ERROR SOURCE CODE
HELP MESSAGE: SOURCE CODE FORMAT IS "Q"

GPA 5
PROMPT: ENTER GPA
ERROR MESSAGE: ERROR GPA
HELP MESSAGE: GPA FORMAT IS "Q"

SCAT/CEH TEST 6
PROMPT: ENTER SCAT/CEH TEST >>>>
ERROR MESSAGE: ERROR TEST
HELP MESSAGE: TEST FORMAT IS "999"

TRANSACTION NAME-----APPLICATION-----LENGTH-----CREATOR-----
HFI HFI 18 01/01/21 GLFMD

-----ELEMENTS-----

UNIT ASSIGNED 10
PROMPT: PLEASE ENTER UNIT ASSIGNED TO
4-POSITION ALPHA-NUMERIC
ERROR MESSAGE: SYNTAX ERROR
THIS FIELD IS 4 CHARACTERS A/N

CADET NUMBER 20

TRANSACTION SCHEM LAYOUT

52

1	2	3	4	5	6	7	8
12345678901234567890123456789012345678901234567890							
EVERY UNIT ASSIGNED			XXXX				
CADET NUMBER?			9999				
NUMBER OF PISH UPS?			999				
HOW MANY BENT LEG SIT UPS?			999				
MILE RUN - 14 SECONDS?			999				
WALVER CODE?			A				
01	02	03	04	05	06	07	08
09	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24

APPLICATION: AIRCRAFTS

6.0 PROCESS II SIMULATION

6.0 PROCESS II SIMULATION

6.1 RUNNING PROCESS II

CASTS Process II executes the transactions specified in PROCESS I. During the execution statistical data are gathered about the effectiveness of the data entry specification and the user interaction. These data are then output in the Terminal Session Summary Report. The user may also specify that a log be maintained of all user interaction with the simulation.

The level of commands are kept to a minimum by allowing Process II to run continuously once a transaction name is entered. Thus, no programming experience is required to run the program. Should the user not know what to do he can ask for help at each step by entering "HELP or "?".

6.1.1 Run Sequence

1. To initiate Process II type in "RUN CASTS2" and hit RETURN.
The program will display an introductory message explaining the program.
2. The user will then be asked to enter his ID. The ID can be up to 9 characters long.
3. The user will then be asked if he wishes to have logging. If the user enters "Y" logging will be initiated, and if he enters "N" there will be no log file kept of the user interaction.
4. The user will then be asked to enter the Input-Output mode desired. He may enter one of the following:

P for PROMPT mode.

In this mode all elements will be prompted individually, a line at a time.

S for SCREEN mode.

In this mode all prompts for a transaction record will appear on the screen at once. The cursor will move from one element input to another as the data is entered correctly.

D for DIRECT mode.

In this mode only one prompt will be issued, asking the user to enter data for an entire transaction record.

B for BATCH Mode.

In this mode there will be no prompting. The data will be entered as a continuous stream from an input device. The input device name will be entered by the user at this time.

5. The user will then be asked to enter the name of the transaction he wishes to execute. The program will look for the transaction name in the data base. If the transaction name is found, the transaction and element information will be loaded into the program and execution will commence. If the program could not find the transaction name in the data base, then the user will be asked to enter another transaction name. If the user cannot locate a transaction in the data base, he should consult his list of transactions. If the user wishes to terminate the session, he may enter "END".

NOTE: In the BATCH mode there will be no prompting for

transaction name. This input will exist in the input device stream.

6. Transaction execution will start and the user interaction will depend on the I/O mode specified.

PROMPT Mode:

- A. The program will display the first element prompt on the screen.
- B. The user will enter the required data.
- C. The program will check the data and depending on the specification from Process I, one of the following sequences could occur
 - i. If the data is good, the data would be accepted and the next prompt issued.
 - ii. If the data is incorrect by -
 - format - a format error would be issued.
 - range - a range error would be issued along with the appropriate ranges.
 - value - a value error would be issued along with a list of legal values.

The user would then receive the same prompt and would input new data. The check process would continue until the data is correct.

- D. The next prompt would be issued and B and C repeated.
- E. When the final element of the transaction is completed, the transaction record would be written to the output file.
- F. The process would then start over with the first element, step A.
- G. If the user wishes to stop this transaction and to go to

another, he may enter "@" at any step.

SCREEN Mode:

- A. The program will display all the transaction record prompts on the screen at one time in a specified format.
- B. The cursor will be positioned at the first element.
- C. The user will enter the required data.
- D. The data will be checked (see PROMPT part C). If the data were incorrect, the error messages would be issued and the cursor repositioned at the same element.
- E. When the data for the element is successfully entered the cursor would move to the next element.
- F. When the final element is completed, the transaction record would be written to the output file.
- G. The process would start over with a screen refresh, step A.
- H. The user may stop the transaction at any time by entering "@".

DIRECT Mode:

- A. The program will display a prompt asking for an entire transaction line to be entered in the correct format and sequence.
- B. The user will enter the transaction data record.
- C. The program will check the data (see PROMPT part C). The first bad element detected will receive an error message and the program will go into the PROMPT mode to prompt for that element. When the user enters the correct data, the program will return to the DIRECT mode to check the next element. This sequence will continue until the entire transaction

record is checked.

- D. When the final element is completed, the transaction record will be written to the output file.
- E. The process would start over with the next record (part A).
- F. The user may stop the transaction by entering "@".

BATCH Mode:

- A. The program will process data from the input device in a steady stream until an @ and END are encountered.
- B. The data will be checked and any transaction record which has bad data will be flagged and discarded.

The input data stream is obtained from the file (or device) given at the beginning of Process II. When "B" (batch) is entered for I/O mode, Process II requests an input file-name and an output file-name. Any error messages are sent to the latter file.

The input data stream must 'look' exactly like the user input would in the DIRECT I/O mode. There must be a card or record containing the name of the transaction for which data is to be entered, followed by the data.

Data input is terminated by the "@" and the last card or record must be an 'END' in the first three positions.

- 7. When the user has terminated the terminal session with the END command, a Terminal Session Summary Report will be written.

6.1.2 Commands

HELP - issues appropriate help
? - issues appropriate help
@ - terminates a transaction
END - terminates a session

6.1.3 Files Produced

The user identification, <user id> entered at the beginning of Process II, is used as the file name for several files produced by Process II. For each file, and appropriate extension is appended. The files are:

<user id>.SRF
<user id>.TDF
<user id>.TLF

Process II automatically produces a Terminal Session Summary Report of the simulation run at the end of the simulation. This is in the file <user id>.SRF. The data entered by the user of the simulation is stored in the transaction data file, <user id>.TDF. A time-stamped log of all of the interactions between the user and the simulation is kept in the transaction log file, <user id>.TLF. Reports can be produced from these files.

6.1.4 Data Base Errors

Should the program encounter a data base error when looking for information specified during Process I and stored in the data base, an error message will be output containing the number ID of the error along with the name and number of the person to contact to correct the error.

6.2 SIMULATION INTERACTION LOG

The simulation log file contains a time stamped log of all of the user interaction with the Process II Simulation. Following Process II, reports can be obtained that display this information. Logging information can be useful in analyzing the effectiveness of the system design as specified in Process I. Problem areas can be identified and the system specifications changed to obviate them.

The user of CASTS Process II is asked whether he would like the log file to be written. A log would not be necessary if Process II is used as a front end data entry device. The overall summary statistics on the terminal session will be produced whether or not logging occurs.

The simulation interaction log records which transactions within which applications were used during the simulation run. It also records all of the terminal I/O, which is the interaction with the user. The different I/O types are prompts, input data, default data, help messages and error messages. When the input data entered for an element is in error, the error is logged. Various error types are; not numeric, not alphabetic, no special characters allowed, special character does not match, invalid value, and not within valid range. The report of all user interaction, i.e., the entire log, is called the Interaction Log Report. The report detailing only the errors that occurred during a Process II Simulation run is the Terminal Session Error Report.

6.3 PROCESS II ERROR MESSAGES

Errors can occur in Process II when raw application data entered does

not conform to the edit criteria specified in Process I. Data are checked position by position against the format specification and the first error found is flagged. The message states the type of error and if applicable, the position where it was encountered.

DATA MUST BE NUMERIC

The character in the position stated must be numeric, that is, a digit from 0-9.

DATA MUST BE ALPHABETIC

The character in the position stated must be alphabetic, A through Z.

NO SPECIAL CHARACTERS ALLOWED

The character in the position stated can be alphabetic, A through Z, or a digit, 0 - 9; but it cannot be a special character. Special characters include period, comma, semi-colon, per cent, etc.

SPECIAL CHARACTER DID NOT MATCH FORMAT

The character in the position stated must be the particular special character that was given in the element format.

DATA MUST BE ONE OF THE FOLLOWING VALUES

The data given for the element satisfied the format specification but was not one of the values given in the value check list of valid values.

DATA MUST FALL WITHIN THE RANGE

The data given for the element satisfied the format specification but failed the range check. It must fall on or within the upper and lower bounds given.

DATA MUST BE A DECIMAL POINT

The character in the position stated must be a decimal point, as specified in the format.

DATA MUST BE A PLUS OR A MINUS SIGN

The character in the position stated must be a numeric sign, either plus (+) or minus (-).

7.0 PROCESS II REPORTS

7.0 PROCESS II REPORTS

A Terminal Session Summary Report is automatically generated at the end of each Process II session. This report provides a statistical summary of the user interaction with the simulation process. It can be obtained by printing the file "<user id>.SRF" subsequent to Process II, where <user id> is the identification given at the beginning of Process II.

In addition to the Terminal Session Summary there are three reports which can be run these reports are:

1. Terminal Session Error Report
2. Interaction Log Report
3. Application Raw Data Report

Each report will run as a separate program. The reports provide performance information about the user interaction with Process II. They are intended to assist the system designer in the evaluation of the specifications and any useful modifications.

7.1 TERMINAL SESSION ERROR REPORT

The Terminal Session Error Report (ERRRPT) contains the errors made by the Process II user during the user interaction with the simulation process. They are reported in alphabetical order by transaction name and chronological order by transaction header time stamp.

The Terminal Session Error Report is initiated by the command:

RUN ERRRPT

This command will load and execute the ERRRPT program. The user interaction sequence is as follows:

1. The user will be asked to enter the ID of the simulation interaction log file (the ID of the person who created the

file).

2. The program will attempt to locate the file. If the file is located processing commences. If the file is not located the user will be asked for another ID. If the user cannot name an existing file he may terminate by typing "END".
3. The program will sort the errors by application and transactions.
4. The report can be obtained by printing the file "<user id>.ERF".

7.2 INTERACTION LOG REPORT

The Interaction Log Report (LOGRPT) is a printed version of the simulation interaction log. It consists of the complete Process II interactive session with time stamps on each user interaction.

The Interaction Log Report is initiated by the command:

RUN LOGRPT

This command will load and execute the LOGRPT program. The user interaction sequence is as follows:

1. The user will be asked to enter the ID of the person who created the file.
2. The program will attempt to locate the file. If the file is located processing commences. If the file is not located the user will be asked for another ID. If the user cannot name an existing file he may terminate by typing "END".
3. The program will then write a report of terminal/user dialogue in the precise sequence of occurrence.
4. The report can be obtained by printing the file "<userid>.LRF".

7.3 APPLICATION RAW DATA REPORT

The Application Raw Data Report (DATRPT) consists of a literal picture of the raw data input for the transactions exactly as input by the user during the simulation.

The Application Raw Data Report is initiated by the command:

RUN DATRPT

This command will load and execute the DATRPT program. The user interaction sequence is as follows:

1. The user will be asked to enter the ID of the application raw data file (the ID of the person who created the file).
2. The program will attempt to locate the file. If the file is located processing commences. If the file is not located the user will be asked for another ID. If the user cannot name an existing file he may terminate by typing "END".
3. The program will then write a report of the transaction record element data as entered by the PROCESS II user.
4. The report can be obtained by printing the file "<user id>.DRF".

COMPUTER AIDED SPECIFICATION TESTING SYSTEM

AIR4ICS CIT

PAGE 1

DATE 8/23/06

TERMINAL SESSION SUMMARY REPORT

TIME IN 17:49:46 TIME OUT 17:58:16

USER ID: UM

APPLICATION : AIR4ICS

TRANSACTIONS USED				*****			
ACC	6	27	6	47	1.31	11	0.31
HFT	8	26	6	58	1.21	10	0.21
ORDER-STICK	9	11	2	18	1.00	7	0.39
3 DIFFERENT TRANSACTIONS ENTERED				-----			
GRAND TOTALS				123	3.52	28	0.91
AVERAGE / TRANSACTION				41	1.17	9	0.30

COMPUTER AIDED SPECIFICATION TESTING SYSTEM

AIRMICS GIT

PAGE 1

DATE 01/03/06

TERMINAL SESSION ERROR REPORT

USER ID: UM

APPLICATION: AGC

TRANSACTION: AGC

1 17:32:06

ELEMENT: UNIT ASSIGNED
ERROR CONDITION: NO SPECIAL CHARACTERS ALLOWED
BAD DATA DUMP: 666

ELEMENT: CADET NUMBER
ERROR CONDITION: DATA MUST BE NUMERIC
BAD DATA DUMP: AX
ERROR CONDITION: DATA MUST BE NUMERIC
BAD DATA DUMP: H90-3

ELEMENT: STATUS
ERROR CONDITION: NO SPECIAL CHARACTERS ALLOWED
BAD DATA DUMP: 6

ELEMENT: SOURCE CODE
ERROR CONDITION: DATA MUST BE NUMERIC
BAD DATA DUMP: Y

APPLICATION: AIRMICS

TRANSACTION: NPT

1 17:21:09

ELEMENT: CADET NUMBER
ERROR CONDITION: DATA MUST BE NUMERIC
BAD DATA DUMP: 3

ELEMENT: PUSH UP
ERROR CONDITION: DATA MUST BE NUMERIC
BAD DATA DUMP: 50

2 17:28:23

ELEMENT: CADET NUMBER

DATE 81/03/06

COMPUTER AIDED SPECIFICATION TESTING SYSTEM

AIRMICS GIT

PAGE 1

INTERACTION LOG REPORT

USER ID: DH

APPLICATION: AIRMICS

TRANSACTION: *****

AGC

TIME	TYPE	TEXT
17:50:58	PROMPTS	CURRENT TRANSACTION : AGC
17:50:59	PROMPTS	PLEASE ENTER UNIT ASSIGNED TO
17:50:59	PROMPTS	4-POSITION ALPHA-NUMERIC
17:50:59	PROMPTS	1234
17:50:59	PROMPTS	ENTER CADET NUMBER(9999)
17:50:59	PROMPTS	1234
17:50:59	PROMPTS	ENTER STATUS
17:50:59	PROMPTS	W
17:50:59	PROMPTS	ENTER SOURCE CODE
17:50:59	PROMPTS	E
17:50:59	PROMPTS	>>> ERROR IN THIS ELEMENT ENTRY, COLUMN 1 -->>>
17:50:59	PROMPTS	ERROR SOURCE CODE
17:50:59	PROMPTS	DATA MUST BE NUMERIC
17:50:59	PROMPTS	ENTER SOURCE CODE
17:50:59	PROMPTS	5
17:50:59	PROMPTS	ENTER GPA
17:50:59	PROMPTS	3
17:50:59	PROMPTS	ENTER SCAT/CEB TEST >>>>
17:50:59	PROMPTS	A
17:50:59	PROMPTS	>>> ERROR IN THIS ELEMENT ENTRY, COLUMN 1 -->>>
17:50:59	PROMPTS	A
17:50:59	PROMPTS	ERROR TEST
17:50:59	PROMPTS	DATA MUST BE NUMERIC
17:50:59	PROMPTS	ENTER SCAT/CEB TEST >>>>
17:50:59	PROMPTS	123
17:50:59	PROMPTS	CURRENT TRANSACTION : AGC
17:50:59	PROMPTS	PLEASE ENTER UNIT ASSIGNED TO
17:50:59	PROMPTS	4-POSITION ALPHA-NUMERIC
17:50:59	PROMPTS	1234
17:50:59	PROMPTS	ENTER CADET NUMBER(9999)
17:50:59	PROMPTS	W
17:50:59	PROMPTS	ENTER STATUS
17:50:59	PROMPTS	W
17:50:59	PROMPTS	ENTER SOURCE CODE

COMPUTER AIDED SPECIFICATION TESTING SYSTEM

PAGE 1

DATE 01/03/76

AIRNICS GIT

APPLICATION RAW DATA REPORT

USER ID: DR

APPLICATION : AIRNICS

TRANSACTION

AGC

LENGTH: 16

12345678901234567890123456789012345678901234567890123456789012345678901

1234123453 123
01001234532 66h
0198900742 70h
7807840432 500
5678889541 456
908 7776732 234

RPT

LENGTH: 1H

12345678901234567890123456789012345678901234567890123456789012345678901

84071234000076300Z
780067890067056400A
84075678000667500S
78903456012013700K
49077890234173500h
78902345050050300A
73452345050050400A
23452345050050500Z

URLLEN-STUCK

LENGTH: 51

12345678901234567890123456789012345678901234567890123456789012345678901

55555555
12345678

1 2